UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/618,118	07/11/2003	Pierluigi Pugliese	Pugliese 31	4099

47384 7590 06/17/2009

RYAN, MASON & LEWIS, LLP
90 FOREST AVENUE
LOCUST VALLEY, NY 11560

EXAMINER	
HICKS, MICHAEL J	

ART UNIT	PAPER NUMBER
2165	

MAIL DATE	DELIVERY MODE
06/17/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/618,118	PUGLIESE, PIERLUIGI	
	Examiner	Art Unit	
	Michael J. Hicks	2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *26 May 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *22-24,26-36,38-40 and 42-45* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *22-24, 26-36, 38-40, and 42-45* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 22-24, 26-36, 38-40, and 42-45 Pending.

Claims 1-21, 25, 30-31, 37, and 41 Canceled.

Response to Arguments

2. Applicant's arguments filed 5/26/2009 have been fully considered but they are not persuasive.

As per Applicants arguments regarding Debevc, Examine respectfully disagrees. Examiner asserts that arguments directed towards Debevc relating to the limitation of "the replacing step is executed only if the calculated difference exceeds a threshold, the threshold being a number of menu items greater to or equal to two " were addressed in the previous Office Action, dated 2/26/2009. While Applicant has reasserted Applicants arguments, Examiner notes that Applicant failed to address the arguments presented by Examiner in the previous Office Action. As such, Examiner will reassert Examiners previously response. Examiner notes that the sections of Debevc cited by Applicants in Applicants arguments clearly indicate that multiple changes are tracked by the system of Debevc, evidenced by the disclosure that the system continues to dynamically calculate the priority of each command. This disclosure clearly indicates that while only one change may be proposed at a time in the system of Debevc, multiple changes may be tracked, therefor allowing the threshold to be considered to be two or greater. In summary, Examiner asserts that while only

one change will be proposed at a time by the system of Debevc, the total number of changes which are being tracked and which may be proposed can be greater than one (e.g. greater to or equal to two).

As per Applicants arguments regarding Hoffberg, Examine respectfully disagrees. Examiner notes that Paragraphs 1144 and 1145 as well as Figure 17 of Hoffberg describe an adaptable menu of choices which are presented to a user for selection/confirmation. Examiner notes that the art of Hoffberg is only intended to cover the limitation that "the menu structure which is displayed for approval is the concurrent display of the entire menu structure". Examiner further notes that Hoffberg is not relied upon to teach the menu being displayed for approval as the rejection clearly states that this step is performed by Schaffer in view of Debevc, but that Hoffberg, in order to disclose the claim limitation in combination with Schaffer and Debevc, must only show that the menu that is displayed is a concurrent display of the entire menu structure. Examiner asserts that as the cited sections of Hoffberg clearly describe an adaptable menu of choices, the determination of the adaptable menu, and the presentation to the user of the adaptable menu in its entirety, the art of Hoffberg clearly anticipates the claim limitation of "the menu structure which is displayed for approval is the concurrent display of the entire menu structure".

In light of the above arguments, the rejection in view of the previously cited references of Schaffer, Debevc, and Hoffberg will be maintained.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 29, 32-33, and 43 rejected under 35 U.S.C. 103(a) as being unpatentable over Schaffer in view of Debevc.

As per Claim 29, Schaffer discloses a processor-implemented method for rearranging a plurality of menu items within a menu structure of a user interface, the method comprising the steps of collecting data about respective selection rates of the menu items within a current menu structure (i.e. *"In one embodiment, the resequencing occurs adaptively and is based upon monitoring selections of the menu items over time. The menu items are typically options in which some options are selected and others remain unselected. Each selection of each option is counted. A resequencing of the menu items is determined by the frequency of selection. Thus, the menu options are adaptively rearranged in a frequency-based order, with the most often selected option being presented first in the next utilization of the user interface."* The preceding text excerpt clearly indicate that data is collected about the frequency of menu selection (e.g. a tracking of the number of times each menu item is selected) prior to adapting the menu structure (e.g. this data would have to be collected in order to determine the frequency of menu items, which is needed to perform the frequency re-ordering from a default configuration, such as shown in Figures 3-5).) (Column 2, Lines 5-16); calculating a new menu structure based on the collected data about the respective selection rates of the menu items within the current menu structure

(i.e. *"In one embodiment, the resequencing occurs adaptively and is based upon monitoring selections of the menu items over time. The menu items are typically options in which some options are selected and others remain unselected. Each selection of each option is counted. A resequencing of the menu items is determined by the frequency of selection. Thus, the menu options are adaptively rearranged in a frequency-based order, with the most often selected option being presented first in the next utilization of the user interface. "* The preceding text excerpt clearly indicate that data is collected about the frequency of menu selection (e.g. a tracking of the number of times each menu item is selected) prior to adapting the menu structure (e.g. this data would have to be collected in order to determine the frequency of menu items, which is needed to perform the frequency re-ordering from a default configuration, such as shown in Figures 3-5).) (Column 2, Lines 5-16); and replacing the current menu structure with the new menu structure (i.e. *"In one embodiment, the resequencing occurs adaptively and is based upon monitoring selections of the menu items over time. The menu items are typically options in which some options are selected and others remain unselected. Each selection of each option is counted. A resequencing of the menu items is determined by the frequency of selection. Thus, the menu options are adaptively rearranged in a frequency-based order, with the most often selected option being presented first in the next utilization of the user interface. "* The preceding text excerpt clearly indicate that data is collected about the frequency of menu selection (e.g. a tracking of the number of times each menu item is selected) prior to adapting the menu structure (e.g. this data would have to be collected in order to determine the frequency of menu items, which is needed to perform the frequency re-ordering from a default configuration, such as shown in Figures 3-5).) (Column 2, Lines 5-16); wherein user approval of menu alteration is obtained via the user interface prior to completion of the replacing step (i.e. *"As another optional feature, the resequencing may be disabled to turn "off" the statistical collection that counts the item selections. In utilizing this feature, the user may invoke a resequence option that initiates rearrangement of the menu items based upon the "learning" that occurred since the adaptation option was last enabled. Allowing adaptation to be enabled and disabled is beneficial*

for those instances in which a user is performing operations that are exceptions to the norm or are single-time activities. As a related optional feature, the statistical collection may remain "on," but with the resequencing occurring only upon the command of the user. This prevents unexpected and/or unwanted resequencing from causing difficulties for the user." The preceding text excerpt clearly indicates that the system may be configured such that the user must initiate the resequencing procedure using a command, thereby approving the menu alteration and utilizing the user interface. Note that this step may take place before the replacing, collecting, and calculating steps.) (Column 2, Lines 25-37).

Schaffer fails to disclose the calculating step further comprises the step of calculating a difference between the new menu structure and the current menu structure, the difference is a number of menu items in the new menu structure that have no corresponding match in the current menu structure, and the replacing step is executed only if the calculated difference exceeds a threshold, the threshold being a number of menu items greater to or equal to two.

Debevc discloses the calculating step further comprises the step of calculating a difference between the new menu structure and the current menu structure (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again.*

Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity" The preceding text excerpt along with Figure 1 clearly indicates that a difference between the current and suggested menus is calculated which prompts the system to indicate to the user that the new menu is available.) (Page 4, Figure 1), the difference is a number of menu items in the new menu structure that have no corresponding match in the current menu structure (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity"* The preceding text excerpt along with Figure 1 and Figure 2

clearly indicates that the difference is identified by identifying an icon which the system feels should be added which has no corresponding match in the current menu structure.) (Page 4, Figures 1-2), and wherein the replacing step is executed only if the calculated difference exceeds a threshold, the threshold being a number of menu items greater to or equal to two (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity"* The preceding text excerpt along with Figure 1 clearly indicates that the user may choose to disregard the predefined threshold and only consider toolbar changes at their convenience, thus indicating that the threshold may be user defined (e.g. the threshold may be set two or more by the user).) (Page 4, Figure 1).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Schaffer with the teachings of

Debevc to include the step of calculating a difference between the new menu structure and the current menu structure, the difference is a number of menu items in the new menu structure that have no corresponding match in the current menu structure, and the replacing step is executed only if the calculated difference exceeds a threshold with the motivation to design an adaptive user interface in a computer environment familiar to many users.

As per Claim 32, Schaffer fails to disclose the threshold is predefined.

Debevc discloses the threshold is predefined (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity"* The preceding text excerpt along with Figure 1 clearly indicates that the threshold is predefined to be a single change to the toolbar.) (Page 4, Figure 1).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Schaffer with the teachings of Debevc to include the threshold is predefined with the motivation to design an adaptive user interface in a computer environment familiar to many users.

As per Claim 33, Schaffer fails to disclose the threshold is selected by the user.

Debevc discloses the threshold is selected by the user (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity"* The preceding text excerpt along with Figure 1 clearly indicates that the user may choose to disregard the predefined threshold and only consider toolbar changes at their convenience, thus indicating that the threshold may be user defined.) (Page 4, Figure 1).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Schaffer with the teachings of Debevc to include the threshold is selected by the user with the motivation to design an adaptive user interface in a computer environment familiar to many users.

As per Claim 43, Schaffer fails to disclose the step of displaying the new menu structure to the user prior to completion of the replacing step, wherein the displaying step is executed only if the calculated difference exceeds a threshold, and wherein the user approval comprises user approval of the new menu structure as displayed.

Debevc discloses the step of displaying the new menu structure to the user prior to completion of the replacing step (i.e. *"At the user's convenience, the adaptive bar offers suggestions for adding or removing command icons, based on the frequency and probability of specific commands. It also implements these changes once the user has agreed to them."* The preceding text excerpt along with Figure 1 clearly indicates that the system displays the new menu structure to the user prior to the completion of the replacing step (e.g. the user must approve the suggested menu changes before they are implemented).) (Abstract), wherein the displaying step is executed only if the calculated difference exceeds a threshold (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single*

dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity" The preceding text excerpt along with Figure 1 clearly indicates that the user may choose to disregard the predefined threshold and only consider toolbar changes at their convenience, thus indicating that the threshold may be user defined (e.g. the threshold may be set two or more by the user).) (Page 4, Figure 1), and wherein the user approval comprises user approval of the new menu structure as displayed (i.e. *"At the user's convenience, the adaptive bar offers suggestions for adding or removing command icons, based on the frequency and probability of specific commands. It also implements these changes once the user has agreed to them."* The preceding text excerpt along with Figure 1 clearly indicates that user approval of the suggested menu changes is obtained after the new suggested menu structure is displayed to he user.) (Abstract).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Schaffer with the teachings of Debevc to include the step of displaying the new menu structure to the user prior to completion of the replacing step, wherein the displaying step is executed only if the calculated difference exceeds a threshold, and wherein the user approval comprises user approval of the new menu structure as displayed with the

motivation to design an adaptive user interface in a computer environment familiar to many users.

5. Claims 22-24, 26-28, 34-36, and 38-40, 42, and 44-45 rejected under 35 U.S.C. 103(a) as being unpatentable over Schaffer in view of Debevc and further in view of Hoffberg et al. (U.S. Pre-Grant Publication Number 2002/0151992 and referred to hereinafter as Hoffberg).

As per Claims 22, 34, and 38, Schaffer discloses a processor-implemented method, device, and machine readable storage medium for rearranging a plurality of menu items within a menu structure of a user interface, the method comprising the steps of collecting data about respective selection rates of the menu items within a current menu structure (i.e. *"In one embodiment, the resequencing occurs adaptively and is based upon monitoring selections of the menu items over time. The menu items are typically options in which some options are selected and others remain unselected. Each selection of each option is counted. A resequencing of the menu items is determined by the frequency of selection. Thus, the menu options are adaptively rearranged in a frequency-based order, with the most often selected option being presented first in the next utilization of the user interface. "* The preceding text excerpt clearly indicate that data is collected about the frequency of menu selection (e.g. a tracking of the number of times each menu item is selected) prior to adapting the menu structure (e.g. this data would have to be collected in order to determine the frequency of menu items, which is needed to perform the frequency re-ordering from a default configuration, such as shown in Figures 3-5).) (Column 2, Lines 5-16); calculating a new menu structure based on the collected data about the

respective selection rates of the menu items within the current menu structure (i.e. *"In one embodiment, the resequencing occurs adaptively and is based upon monitoring selections of the menu items over time. The menu items are typically options in which some options are selected and others remain unselected. Each selection of each option is counted. A resequencing of the menu items is determined by the frequency of selection. Thus, the menu options are adaptively rearranged in a frequency-based order, with the most often selected option being presented first in the next utilization of the user interface. "* The preceding text excerpt clearly indicate that data is collected about the frequency of menu selection (e.g. a tracking of the number of times each menu item is selected) prior to adapting the menu structure (e.g. this data would have to be collected in order to determine the frequency of menu items, which is needed to perform the frequency re-ordering from a default configuration, such as shown in Figures 3-5).) (Column 2, Lines 5-16); and replacing the current menu structure with the new menu structure (i.e. *"In one embodiment, the resequencing occurs adaptively and is based upon monitoring selections of the menu items over time. The menu items are typically options in which some options are selected and others remain unselected. Each selection of each option is counted. A resequencing of the menu items is determined by the frequency of selection. Thus, the menu options are adaptively rearranged in a frequency-based order, with the most often selected option being presented first in the next utilization of the user interface. "* The preceding text excerpt clearly indicate that data is collected about the frequency of menu selection (e.g. a tracking of the number of times each menu item is selected) prior to adapting the menu structure (e.g. this data would have to be collected in order to determine the frequency of menu items, which is needed to perform the frequency re-ordering from a default configuration, such as shown in Figures 3-5).) (Column 2, Lines 5-16); wherein user approval of menu alteration is obtained via the user interface prior to completion of the replacing step (i.e. *"As another optional feature, the resequencing may be disabled to turn "off" the statistical collection that counts the item selections. In utilizing this feature, the user may invoke a resequence option that initiates rearrangement of the menu items based upon the "learning" that occurred since the*

adaptation option was last enabled. Allowing adaptation to be enabled and disabled is beneficial for those instances in which a user is performing operations that are exceptions to the norm or are single-time activities. As a related optional feature, the statistical collection may remain "on," but with the resequencing occurring only upon the command of the user. This prevents unexpected and/or unwanted resequencing from causing difficulties for the user." The preceding text excerpt clearly indicates that the system may be configured such that the user must initiate the resequencing procedure using a command, thereby approving the menu alteration and utilizing the user interface. Note that this step may take place before the replacing, collecting, and calculating steps.) (Column 2, Lines 25-37).

Schaffer fails to disclose the limitations of the method further comprising the step of concurrently displaying the entire new menu structure to the user prior to completion of the replacing step; and wherein the user approval comprises user approval of the new menu structure as displayed.

Debevc discloses the method further comprising the step of displaying the new menu structure to the user prior to completion of the replacing step (i.e. *"At the user's convenience, the adaptive bar offers suggestions for adding or removing command icons, based on the frequency and probability of specific commands. It also implements these changes once the user has agreed to them."* The preceding text excerpt along with Figure 1 clearly indicates that the system displays the new menu structure to the user prior to the completion of the replacing step (e.g. the user must approve the suggested menu changes before they are implemented).) (Abstract); and wherein the user approval comprises user approval of the new menu structure as displayed (i.e. *"At the user's convenience, the adaptive bar offers suggestions for adding or removing command icons, based on the frequency and probability of specific commands. It also implements these changes once the user has agreed to them."* The preceding text excerpt along with Figure 1 clearly indicates that user approval of the

suggested menu changes is obtained after the new suggested menu structure is displayed to he user.) (Abstract).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Schaffer with the teachings of Debevc to include the method further comprising the step of displaying the new menu structure to the user prior to completion of the replacing step; and wherein the user approval comprises user approval of the new menu structure as displayed with the motivation to design an adaptive user interface in a computer environment familiar to many users.

Hoffberg discloses that the menu structure which is displayed for approval is the concurrent display of the entire menu structure (See Hoffberg, Figure 17, wherein the complete altered menu structure is displayed to the user for approval.).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Schaffer and Debevc with the teachings of Hoffberg to include that the menu structure which is displayed for approval is the concurrent display of the entire menu structure with the motivation of predicting a desired user function, based on user history, as well as machine internal status and context (Hoffberg, Abstract).

As per Claims 23, 35, and 39, Schaffer discloses the user approval is obtained prior to completion of the collecting step (i.e. *"As another optional feature, the resequencing may be disabled to turn "off" the statistical collection that counts the item selections. In utilizing this feature, the user may invoke a resequence option that initiates*

rearrangement of the menu items based upon the "learning" that occurred since the adaptation option was last enabled. Allowing adaptation to be enabled and disabled is beneficial for those instances in which a user is performing operations that are exceptions to the norm or are single-time activities. As a related optional feature, the statistical collection may remain "on," but with the resequencing occurring only upon the command of the user. This prevents unexpected and/or unwanted resequencing from causing difficulties for the user." The preceding text excerpt clearly indicates that the system may be configured such that the user must initiate the resequencing procedure using a command, thereby approving the menu alteration. Note that this step may take place before the replacing, collecting, and calculating steps.) (Column 2, Lines 25-37).

As per Claims 24, 36, and 40, Schaffer discloses the user approval is obtained prior to completion of the calculating step (i.e. *"As another optional feature, the resequencing may be disabled to turn "off" the statistical collection that counts the item selections. In utilizing this feature, the user may invoke a resequence option that initiates rearrangement of the menu items based upon the "learning" that occurred since the adaptation option was last enabled. Allowing adaptation to be enabled and disabled is beneficial for those instances in which a user is performing operations that are exceptions to the norm or are single-time activities. As a related optional feature, the statistical collection may remain "on," but with the resequencing occurring only upon the command of the user. This prevents unexpected and/or unwanted resequencing from causing difficulties for the user."* The preceding text excerpt clearly indicates that the system may be configured such that the user must initiate the resequencing procedure using a command, thereby approving the menu alteration. Note that this step may take place before the replacing, collecting, and calculating steps.) (Column 2, Lines 25-37).

As per Claim 26, Schaffer discloses the user approval comprises the selection of a specified menu item (i.e. *"As another optional feature, the resequencing may be disabled to turn "off" the statistical collection that counts the item selections. In utilizing this feature, the user may invoke a resequence option that initiates rearrangement of the menu items based upon the "learning" that occurred since the adaptation option was last enabled. Allowing adaptation to be enabled and disabled is beneficial for those instances in which a user is performing operations that are exceptions to the norm or are single-time activities. As a related optional feature, the statistical collection may remain "on," but with the resequencing occurring only upon the command of the user. This prevents unexpected and/or unwanted resequencing from causing difficulties for the user."* The preceding text excerpt clearly indicates that the resequencing takes place in response to a command, which is linked to an option on a menu.) (Column 2, Lines 25-37).

As per Claim 27, Schaffer discloses the menu items are arranged within a plurality of functional groupings within the current menu structure (i.e. *"Second-level menu items are preferably also tracked for frequency of selection. That is, if selection of a particular option in the main menu initiates display of submenu items related to the initial selection, there preferably is a monitoring of the user selection of the submenu items, so that an adaptive frequency-based reordering also occurs at the submenu level."* The preceding text excerpt clearly indicates that menus may include submenus (e.g. functional groupings of commands within the menu structure).) (Column 2, Lines 38-44) and wherein the new menu structure comprises rearrangement of particular ones of the menu items within at least a given one of the functional groupings while maintaining said plurality of functional groupings of the menu items (i.e. *"Second-level menu items are preferably also tracked for frequency of selection. That is, if selection of a particular option in the main menu initiates display of submenu items related to the initial selection, there preferably is a monitoring*

of the user selection of the submenu items, so that an adaptive frequency-based reordering also occurs at the submenu level." The preceding text excerpt clearly indicates that the submenus (e.g. functional groupings) may be resequenced regarding frequency, while maintaining their structure.) (Column 2, Lines 38-44).

As per Claim 28, Schaffer discloses the functional groupings comprise submenus displayed responsive to the selection of at least one menu item (i.e. *"Second-level menu items are preferably also tracked for frequency of selection. That is, if selection of a particular option in the main menu initiates display of submenu items related to the initial selection, there preferably is a monitoring of the user selection of the submenu items, so that an adaptive frequency-based reordering also occurs at the submenu level."* The preceding text excerpt clearly indicates that menus may include submenus (e.g. functional groupings of commands within the menu structure) which are displayed responsive to the selection of a primary menu item.) (Column 2, Lines 38-44).

As per Claims 42, 44, and 45, Schaffer fails to disclose calculating a difference between the new menu structure and the current menu structure, the difference is a number of menu items in the new menu structure that have no corresponding match in the current menu structure, and the displaying step is executed only if the calculated difference exceeds a threshold, the threshold being a number of menu items greater to or equal to two.

Debevc discloses the calculating step further comprises the step of calculating a difference between the new menu structure and the current menu structure (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the*

background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity" The preceding text excerpt along with Figure 1 clearly indicates that a difference between the current and suggested menus is calculated which prompts the system to indicate to the user that the new menu is available.) (Page 4, Figure 1), the difference is a number of menu items in the new menu structure that have no corresponding match in the current menu structure (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion*

over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's most recent activity" The preceding text excerpt along with Figure 1 and Figure 2 clearly indicates that the difference is identified by identifying an icon which the system feels should be added which has no corresponding match in the current menu structure.) (Page 4, Figures 1-2), and wherein the displaying step is executed only if the calculated difference exceeds a threshold, the threshold being a number of menu items greater to or equal to two (i.e. *"The most important feature of the adaptive bar is its ability to guide and automate the process of adding and removing icons from the toolbar. Whenever the system determines that a change to the bar may be appropriate, it plays a tone and changes the background color of the bar. (The particular color to which the bar changes can be customized.) Once the bar background indicates that a proposal for change is available, the user can review the proposal at any time by double-clicking on the bar background. This action calls up a single dialog box (Figure 2) that allows the user to confirm or reject the proposed change. If the user rejects a proposed change, the system maintains the data that led to the suggestion, but then uses this data to generate different proposals that have not yet been rejected. This mechanism helps prevent the system from insisting on one particular suggestion over and over again. Because only the background color changes when there is a suggestion, the user need not stop working and can control when and how the bar is changed. If the user keeps working without reviewing the proposed change, the bar simply retains the new color. If the user continues working for a long time without reviewing any proposals for change, the system continues to dynamically calculate the priority of each command. If at some later time the user clicks on the bar background to review a proposal, the system presents a single proposal based on the user's*

most recent activity" The preceding text excerpt along with Figure 1 clearly indicates that the user may choose to disregard the predefined threshold and only consider toolbar changes at their convenience, thus indicating that the threshold may be user defined (e.g. the threshold may be set two or more by the user).) (Page 4, Figure 1).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Schaffer with the teachings of Debevc to include the step of calculating a difference between the new menu structure and the current menu structure, the difference is a number of menu items in the new menu structure that have no corresponding match in the current menu structure, and the displaying step is executed only if the calculated difference exceeds a threshold with the motivation to design an adaptive user interface in a computer environment familiar to many users.

Conclusion

6. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will

the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Points of Contact

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Hicks whose telephone number is (571) 272-2670. The examiner can normally be reached on Monday - Friday 9:00a - 5:30p.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Neveen Abel-Jalil can be reached at (571)272-4074. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Michael J Hicks
Art Unit 2165
Phone: (571) 272-2670
Fax: (571) 273-2670

/Neveen Abel-Jalil/
Supervisory Patent Examiner, Art Unit 2165